

AMENDMENTS TO THE SPECIFICATION

Please amend the specification as follows:

On page 4 please replace the second (2nd) paragraph, which extends between lines 12-18, with the following paragraph:

A1
Upon installation, a USB device typically provides an operating system with USB standard class and subclass ~~codes~~ codes, which are used to determine whether a generic, or default device driver can be used to control the device. If so, a special OEM/IHV supplied device driver may not be necessary to control the device. Thus, it is often unnecessary for customers to install the device drivers provided ~~on~~ by the installation media for the device to function properly upon being attached.

On page 5, please replace the second (2nd) paragraph positioned between lines 4-16 with the following paragraph:

A2
One of the benefits of using a USB device is ~~the reduced~~ to reduce the amount of interaction typically required of a user to attach and configure a device. This ease of use has typically reduced the amount of device installation related customer support that OEMs/IHVs have needed to provide. Thus, when a default device driver can be used to control a device, it would be beneficial (both in terms of customer ease of use and in terms of the amount of customer support typically required) if installation media such as installation disks or setup programs were

not required to be distributed to specify device-specific UI information. Rather, an operating system could query a device for this additional device specific information. Unfortunately, because host-specific USB requests are not provided or considered by the USB specification, there are no standards that allow a vendor to provide additional USB device-specific information in a USB device in a format that is determined by an operating system.

On page 6, please delete line 9, which is a blank line.

On page 8 please replace the last paragraph, which extends to page 9, lines 1-8, with the following paragraph:

The peripheral device 114 responds to requests from the host computer 102 across the communication path 128. These requests are made using control transfers where setup packets (not shown) are exchanged. The USB device returns descriptors in response to exchanging such setup packets. Although the USB Specification defines a number of different standard, class and vendor specific descriptors, an extended property descriptor 122 is not defined in the USB specification. The extended property descriptor includes UI information that pertains to the peripheral device. The UI information can be in any format such as a format specified by an operating system vendor. The extended property descriptor allows OEMs/IHVs to store device specific UI information such as ~~store~~ icons, fonts, pictures, labels, help pages, Universal Resource Locator (URL) Internet links, and the like, in non-volatile memory 118 of the device.